Epson E1007 4x12 LCD Display Connector Pin Out and Start-up Code

Epson E1007 Description

The Epson E1007 is a 4x12 Monochrome LCD almost identical in operation to the ECM-A1010 display – controller is Epson SED1230 chip. The E1007 has a serial interface while the E1010 uses an 8bit parallel interface. The E1007 is illuminated by four LEDs wired in series (I think) while the E1010 has two. A complete technical description and programming guide for the SED1230 chip is provided in the Epson SED1200 Series Technical User Guide (see the Epson Electronics web site). The E1007 pin out is on the left while the E1010 is on the right. The E1010 has symbols on the top row and left column while the E1007 has fewer top row symbols and no left column symbols. Symbol display control commands differ between the two displays. The spec sheet states that the display will operate from 2.4 to 3.6V – this we found to be true – this is not a 2.8V only display but runs fine from 2.4 to 3.6V.

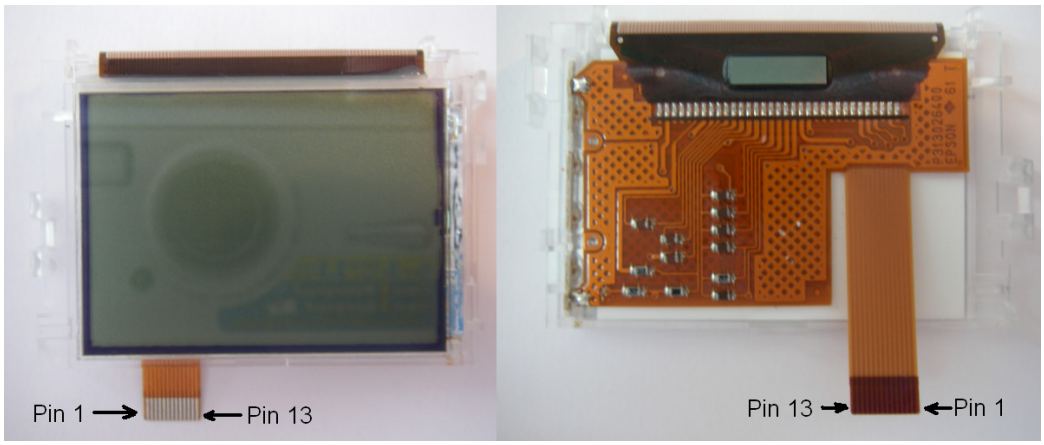
E1007 Pin Assignment

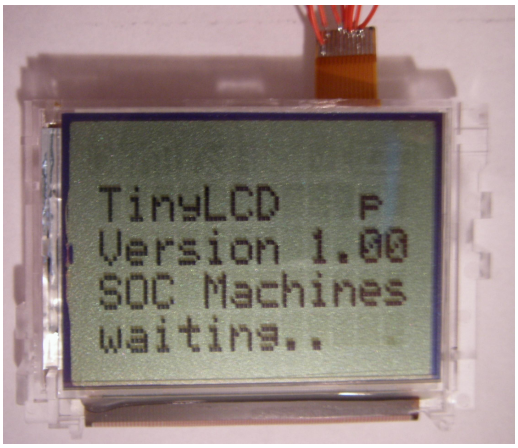
Pin	Symbol	I/O	Description
1	LED+	I	LED Anode
2	LED-	I	LED Cathode
3	A0	I	Selects Register (L-Command, H-Date)
4	CS	I	Chip Select (L-Enable)
5	SI	I	Serial Data
6	SCL	I	Serial Clock
7	VDD	I	Power supply for logic (2.4V-3.6V)
8	GND	I	Ground
9	NC		No Connection
10	NC		No Connection
11	GND2	I	Ground 2
12	VDD2	I	Power supply2 for logic (2.4V-3.6V)
13	RES	I	Reset

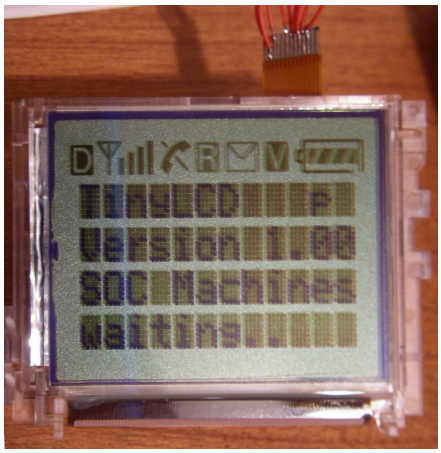
E1010 pin assignment

Pin No.	Symbol	I/O	Function
1	LED1+	I	LED1 Anode
2	LED1-	I	LED1- Cathode
3	A0	I	Selects Register ("L" = Command, "H" = Data)
4	WR	I	Write Enable Input ("L" = Active)
5	CS	I	Chip Selects ("L" = Enable)
6 ~ 13	D7 ~ D0	I	Data Bus
14	VDD	I	Power Supply for Logic(2.8V±1%)
15	VSS	I	Ground
16	RES	I	Reset
17	LED2+	I	LED2 Anode
18	LED2-	I	LED2 Cathode

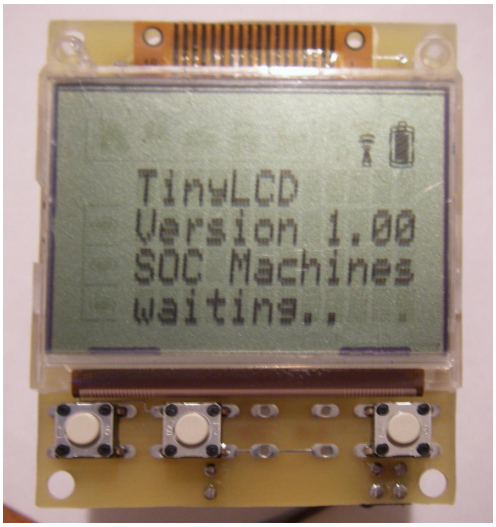
The picture below shows the pin locations described in the table above.

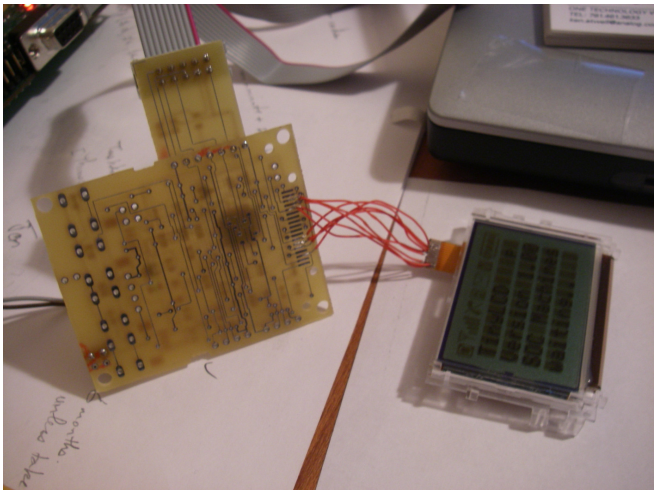






The E1007 display is shown above. The Voltage Volume command is set to 0A on the left and 0F (maximum setting) on the right at 2.8V – note the display washes out at the maximum setting. The E1010 display below with Voltage Volume set to 00 (minimum) – this display washes out at 2.9V – haven't found a way to get this display to work at 3 or 3.3V.



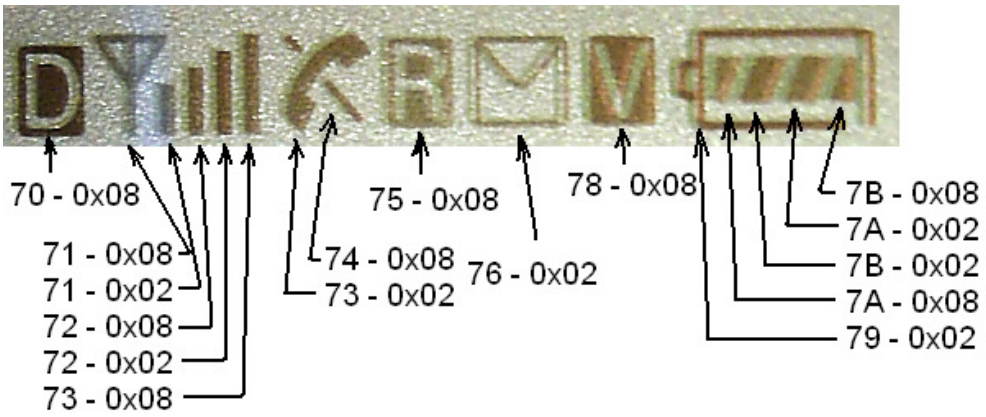


The E1010 display is shown above with some of the top row symbols turned on. The E1010 symbol inventory is more extensive than the E1007. The picture on the right shows our TinyLCD controller driving the E1007 (TinyLCD uses an Atmel ATmega168 processor on the back of the PCB with the E1007 mounted on the front). The display runs just fine from 2.4 to 3.6V – the Voltage Volume setting should be adjusted for correct contrast.

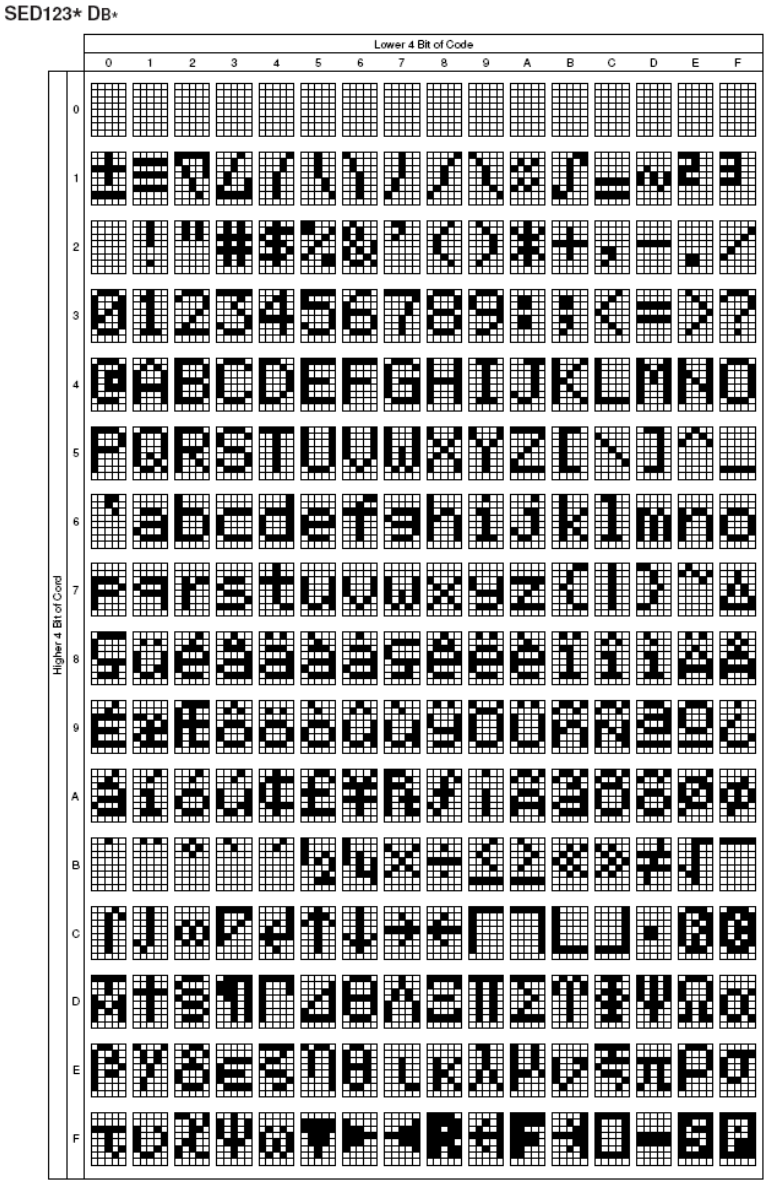
The SED1230 symbol activation is controlled via a 64 bit symbol register. By turning various bits on or off top row symbols can be individually turned on or off. E1010 symbol-register mapping is described in the ECMA1010 User Guide. The E1007 symbol register mapping is shown on the next page.

Character pattern is SED1230-Db (see page 6-27). Nice little display.

Mating connector is Omron XF2H-1315-1 Digikey Part No. OR641CT-ND (haven't tried this connector yet but should work).



Mapping of symbol register to symbol address is given above – some symbols share the same address – all values in hex. E1007 character set is below.



Example C code to program E1007 – note this is a code snippet – not a full executable program – port pin assignment is for an ATmega168 – change accordingly for your target.

```
//**************************************
// E1007 Setup
//**************************************

// Clear display - move cursor to home
#define    LCDCLR   0x01

// move cursor to home position command
#define    LCDHOME         0x10

// Static Display Command
//   D1/0 - 0,0  Display OFF
//   D1/0 - 0,1  1-2Hz blink
//   D1/0 - 1,0  3-4Hz blink
//   D1/0 - 1,1  Display ON
#define    LCDSTATICDISPLAY 0x20
#define    LCDDISPLAYON     0x03
#define    LCDDISPLAYOFF    0x00

// Display control command
//   D3-C, D2-B, D1-DC, D0-D
//   C  -> 1 cursor on, 0 off
//   DC -> 1 double cursor on, 0 off
//   B  -> 1 blink on, 0 off
//   D  -> 1 display on, 0 off
#define    LCDDISPLAY          0x30
#define    LCDCURSORON         0x08
#define    LCDCURSOROFF        0x00
#define    LCDCURSORBLINK      0x04
#define    LCDCURSONNOBLINK    0x00
#define    LCDCURSORDISPLAYON  0x01
#define    LCDCURSORDISPLAYOFF 0x00

// Power Save Command
//   D1-O, D0-PS
//   PS -> 1 power save on, 0 off
//   O  -> 1 oscillator circuit on, 0 off
#define    LCDPOWERSAVE    0x40
#define    LCDPOWERSAVEON  0x01
#define    LCDPOWERSAVEOFF 0x00
#define    LCDOSCON        0x02
#define    LCDOSCOFF       0x00

// Power supply control
//   D2-VC, D1-VF, D0-P
//   VC -> 1 voltage regulator on, 0 off
//   VF -> 1 voltage follower on, 0 off
//   P  -> 1 voltage booster on, 0 off
#define    LCDPOWERSUPPLY   0x50
#define    LCDVOLTREGON     0x04
#define    LCDVOLTREGOFF    0x00
#define    LCDVOLTFOLLOWON  0x02
#define    LCDVOLTFOLLOWOFF 0x00
#define    LCDVOLTBOOSTON   0x01
#define    LCDVOLTBOOSTOFF  0x00

// System set command
```

```
//   D3-N2, D2-N1, D0-CG
//   N2/1 -> 0,0  2 lines
//   N2/1 -> 0,1  3 lines
//   N2/1 -> 1,0  4 lines
//   CG   -> 1 CGRAM used, 0 not used
#define    LCDSYSTEMSET  0x60
#define    LCD4LINE      0x08
#define    LCDCGON       0x01

// Electronic volume setting
//   D3-0 - sets display brightness
#define    LCDVOLREG 0x70

// Set address of DDRAM, CGRAM or symbol register
#define    LCDRAMADDRESS  0x80

//  LCD chip select is active low
#define    CS       0
//  LCD Command/Data  0 - command, 1 data
#define    LA0      2
//  LCD reset line - active high to reset
#define    LCDRESET 3

#define    LCDCONTROL PORTC
#define    LCDDATA    PORTB
#define    LINE1      0x30
#define    LINE2      0x40
#define    LINE3      0x50
#define    LINE4      0x60

void   LCDWriteSerialByte(unsigned char inchar)
{
       unsigned char i,mask;

       mask = 0x80;
       for(i=0; i<8; i++) {
           ClrBit(PORTB,PINB6);
              if((inchar&mask)) SetBit(PORTB,PINB7);
              else ClrBit(PORTB,PINB7);
              mask >>= 1;
              SetBit(PORTB,PINB6);
           }
}

void   LCDprintchar(unsigned char inchar)
{
       LCDCONTROL |= BIT(LA0); // raise A0 high
       LCDCONTROL &= ~BIT(CS); // set CS low
       ShortDelay(1);
       LCDWriteSerialByte(inchar);
       LCDCONTROL |= BIT(CS); // release CS
       ShortDelay(2);
}

void   LCDwritecommand(unsigned char inchar)
{
       LCDCONTROL &= ~BIT(LA0); /* set A0 low */
       LCDCONTROL &= ~BIT(CS);  /* set RS low */
       ShortDelay(1);
       LCDWriteSerialByte(inchar);
       LCDCONTROL |= BIT(CS); // release CS
       ShortDelay(2);
}
```

```
void   LCDDisplayChar(unsigned char address, unsigned char inchar)
{
       LCDwritecommand(LCDRAMADDRESS|address);
       LCDprintchar(inchar);
}


/****************************************
  LCD interface control logic for E1007:
       CS -  0 select device
       LA0 - 0 command, 1 data
 ****************************************/
void   Init_LCD()
{
     SetBit(PORTB,PINB6);  // set clock high
     // Reset LCD - hold in reset for 10usec
     LCDCONTROL &= ~BIT(LCDRESET);  // Set Reset line low
     ShortDelay(200);
     LCDCONTROL |= BIT(LCDRESET);  // Set Reset line high
     ShortDelay(50);

     // Static Icon off - display on - RF circuit on - clear RAM
     // Set display volume

     // System set command
     //   D3-N2, D2-N1, D0-CG
     //   N2/1 -> 0,0  2 lines
     //   N2/1 -> 0,1  3 lines
     //   N2/1 -> 1,0  4 lines
     //   CG   -> 1 CGRAM used, 0 not used
     LCDwritecommand(LCDSYSTEMSET|LCD4LINE|LCDCGON);

     // Static Display Command
     //   D1/0 - 0,0  Display OFF
     //   D1/0 - 0,1  1-2Hz blink
     //   D1/0 - 1,0  3-4Hz blink
     //   D1/0 - 1,1  Display ON
     LCDwritecommand(LCDSTATICDISPLAY|LCDDISPLAYON);

     // Display control command
     //   D3-C, D2-B, D1-DC, D0-D
     //   C  -> 1 cursor on, 0 off
     //   DC -> 1 double cursor on, 0 off
     //   B  -> 1 blink on, 0 off
     //   D  -> 1 display on, 0 off
     LCDwritecommand(LCDDISPLAY|LCDCURSORDISPLAYON);

     // Power Save Command
     //   D1-PS, D0-O
     //   PS -> 1 power save on, 0 off
     //   O  -> 1 oscillator circuit on, 0 off
     LCDwritecommand(LCDPOWERSAVE|LCDPOWERSAVEOFF|LCDOSCON);

     // Set data pointer to DDRAM
     LCDwritecommand(LCDRAMADDRESS|0x30);

     // Wait 20ms for power supply to stabilize
     ShortDelay(30000);

     // Power supply control
     //   D2-VC, D1-VF, D0-P
     //   VC -> 1 voltage regulator on, 0 off
```

```
      //   VF -> 1 voltage follower on, 0 off
      //   P  -> 1 voltage booster on, 0 off
LCDwritecommand(LCDPOWERSUPPLY|LCDVOLTREGON|LCDVOLTFOLLOWON|LCDVOLTBOOSTON);

      // Electronic volume setting
      //   D3-0 - set display brightness
      // TBD - real VDD and adjust Voltage Volume accordingly
      LCDwritecommand(LCDVOLREG|0x0f);

      // move cursor to home position command
      LCDwritecommand(LCDHOME);

}
```